EV257330132US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Liqun CHEN, et al.	RE: Claim to Priority
Serial No.: Not yet assigned	Our Ref: B-5153 621074-2
Filed: Concurrently herewith	
For: "METHOD AND APPARATUS FOR USE IN RELATION TO VERIFYING AN ASSOCIATION BETWEEN TWO PARTIES"	Date: July 2, 2003

CLAIM TO PRIORITY UNDER 35 U.S.C. 119

Mail Stop Patent Application
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

[X] Applicants hereby make a right of priority claim under 35 U.S.C. 119 for the benefit of the filing date(s) of the following corresponding foreign application(s):

COUNTRY	FILING DATE	SERIAL NUMBER
GB	5 July 2002	0215590.1

[ ] A certified copy of each of the above-noted patent applications was filed with the Parent Application No.__.

[X] To support applicants' claim, a certified copy of the above-identified foreign patent application is enclosed herewith.

[ ] The priority document will be forwarded to the Patent Office when required or prior to issuance.

Respectfully submitted,

Richard P. Berg
Attorney for Applicant
Reg. No. 28,145

LADAS & PARRY
5670 Wilshire Boulevard
Suite 2100
Los Angeles, CA 90036
Telephone: (323) 934-2300
Telefax: (323) 934-0202

EV 257330132US

PATENTS • DESIGNS • COPYRIGHT • TRADE MARKS
The Patent Office

INVESTOR IN PEOPLE

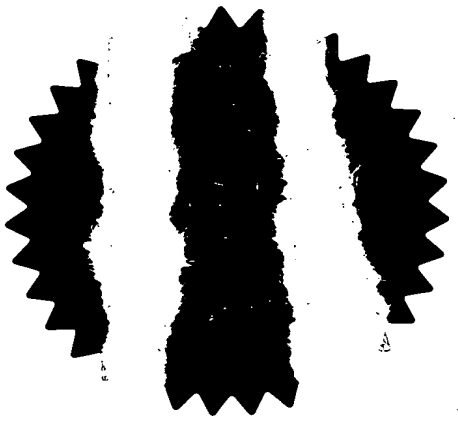
The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.



Signed

Dated 28 August 2002

Patents Form 1/77

Patents Act 1977
(R 6)

THE PATENT OFFICE
G
05 JUL 2002
NEWPORT

The Patent Office

05JUL02 E731256-1 D01463
P01/7700 0.00-0215590.1

1/77

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

The Patent Office

Cardiff Road
Newport
South Wales
NP10 8QQ

1. Your reference — 300202699-01 GB

2. Patent application number *(The Patent Office will fill in this part)* — 0215590.1 — -5 JUL 2002

3. Full name, address and postcode of the or of each applicant *(underline all surnames)*

   Hewlett-Packard Company
   3000 Hanover Street
   Palo Alto
   CA 94304, USA

   Patents ADP number *(if you know it)* 4965880001

   If the applicant is a corporate body, give the country/state of its incorporation — Delaware, USA

4. Title of the invention — Method And Apparatus For Generating A Cryptographic Key

5. Name of your agent *(if you have one)*

   "Address for service" in the United Kingdom to which all correspondence should be sent *(including the postcode)*

   Chris Harrison
   Hewlett-Packard Ltd, IP Section
   Filton Road, Stoke Gifford
   Bristol BS34 8QZ

   Patents ADP number *(if you know it)* 8191439001

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and *(if you know it)* the or each application number

Country	Priority application number *(if you know it)*	Date of filing *(day / month / year)*

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application	Date of filing *(day / month / year)*

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? *(Answer 'Yes' if:*

   a) any applicant named in part 3 is not an inventor, or
   b) there is an inventor who is not named as an applicant, or
   c) any named applicant is a corporate body.
   See note (d))

   Yes

9. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form	
Description	17
Claim(s)	3
Abstract	1
Drawing(s)	2+2

10. If you are also filing any of the following, state how many against each item.

Priority documents	
Translations of priority documents	
Statement of inventorship and right to grant of a patent *(Patents Form 7/77)*	1
Request for preliminary examination and search *(Patents Form 9/77)*	1
Request for substantive examination *(Patents Form 10/77)*	
Any other documents *(please specify)*	Fee Sheet

11. I/We request the grant of a patent on the basis of this application.

Signature [signature] Date 4/7/2002

12. Name and daytime telephone number of person to contact in the United Kingdom: Tony Judd Tel: 0117-312-8026

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

a) If you need help to fill in this form or you have any questions, please contact the Patent Office on 08459 500505.

b) Write your answers in capital letters using black ink or you may type them.

c) If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.

d) If you have answered 'Yes' Patents Form 7/77 will need to be filed.

e) Once you have filled in the form you must remember to sign and date it.

f) For details of the fee and ways to pay please contact the Patent Office.

METHOD AND APPARATUS FOR GENERATING A CRYPTOGRAPHIC KEY

The present invention relates to a method and apparatus for generating a cryptographic key.

With the ever-increasing spread of electronic communication and electronic identification there has been a corresponding increase in demand for cryptographic processes, where users require cryptographic processes to enable encryption of data for security purposes and/or for the purposes of providing identification.

Typically encryption keys are certified by trusted authorises and disseminated using digital certificates where, to allow wide spread availability of cryptographic processes, a hierarchy of trust authorities exist. Within a hierarchy of trust authorities a root trust authority issues a digital certificate to a private/public key to a second level trust authority by using the root authorities private key to sign the second level's trust authorities public key and thereby providing confirmation that the second level private key is authorized by the root authority. Correspondingly the second level trust authority issues a digital certificate to a different private/public key to a third level trust authority that is signed with the second level's private key and so forth. However, for a user to determine that the public key associated with the third level trust authority is derived with the authority of the root trust authority it is necessary for the user to trace the digital certificates that incorporated the various public keys.

It is desirable to improve this situation.

In accordance with a first aspect of the present invention there is provided a method for generating a private key comprising generating a first and second cryptographic key for a first party; generating a third and fourth cryptographic key for a second party wherein the fourth cryptographic key is derived from the first and third cryptographic key; generating a number that in association with the second cryptographic key, the third cryptographic key and the fourth cryptographic key define a first cryptographic parameter, a second cryptographic parameter and a third cryptographic parameter respectively; combining the number with a third party's public key to define an associated private key.

In accordance with a second aspect of the present invention there is provided a method for generating a cryptographic key comprising generating a first cryptographic key and a second cryptographic key for a first party; generating a third cryptographic key and fourth cryptographic key for a second party wherein the fourth cryptographic key is derived from the first cryptographic key and third cryptographic key; generating a number that in association with the second cryptographic key, the third cryptographic key and the fourth cryptographic key define a first, second and third cryptographic parameter respectively; combining the number with a fifth cryptographic key associated with a third party to define an associated cryptographic key such that an association can be established between the fifth cryptographic key of the third party and the second cryptographic key of the first party.

This provides the advantage of allowing a trust authority in one level of a trust hierarchy, given a master private key generated by a trust authority in a higher level of the hierarchy, to generate a private/public key pair without further interaction from the trust authority in the higher level of the hierarchy. This also provides the advantage of allowing a trust hierarchy to be established without requiring the use of digital certificates. Further the public key corresponding to the private key generated by the trust authority can be universally verified where a verifier can be certain that the private key must

have been generated with knowledge of the higher level trust authorities private key and without requiring disclosure of the higher level trust authorities private key.

In accordance with a third aspect of the present invention there is provided a method for generating a private key comprising generating a first private key and public key for a first party; generating a second private and public key for a second party wherein the second private key is derived from the first private key and second public key; generate a number that in association with the first pubic key, the second private and public key define a first, second and third public parameter respectively; combining the number with a third pubic key associated with a third party to define an associated private key such that an association can be established between the third public key of the third party and the first public key of the first party.

Preferably the number is a random number.

Preferably the association between the third public key and first public key is established using a bilinear map, such as a Tate or Weil pairing.

Preferably the first party is a first trusted party and the second party is a second trusted party.

In accordance with a fourth aspect of the present invention there is provided a method for generating a private key comprising generating a first private key and public key for a first party; generating a second private and public key for a second party wherein the second private key is derived from the first private key and second public key; generate a third private key for the second party that in association with the first public key, the second private key and the second public key define a first cryptographic parameter, a second cryptographic parameter and a third public key respectively; combining the third private key with a third party's public key to define an associated private

key such that an association can be established between the third public key of the second party and the first public key of the first party.

In accordance with a fifth aspect of the present invention there is provided a computer apparatus for generating a private key comprising a processor arranged to generate a number that in association with a first private key and public key associated with a first party define a first and second public parameter respectively wherein the first private key is derived from a second private key associated with a second party and the first public key; and combining the number with a second public key associated with a third party to define an associated private key such that an association can be established between the second public key of the third party and a third public key of the second party.

Preferably the association between the second public key and the third public key is established using a bilinear map, such as a Tate or Weil pairing.

Preferably the first party is a first trusted party and the second party is a second trusted party.

For a better understanding of the present invention and to understand how the same may be brought into effect reference will now be made, by way of example only, to the accompanying drawings, in which:-

Figure 1 illustrates a computer system according to an embodiment of the present invention;

Figure 2 illustrates a computer system according to an embodiment of the present invention.

Figure 1 shows a first computer entity 10, a second computer entity 20, a third computer entity 30 and a fourth computer entity 40 connected via a network 50, for example the Internet.

The first computer entity 10 represents a first trust authority 60, for example a company, the second computer entity 20 represents a second trust authority 70, for example a division within the company and the third computer entity 30 represents a user 80, for example a worker within the company. The fourth computer entity 40 represents, for example, a business partner 90 of the company that wishes to interact with the user 80.

The first, second, third and fourth computer entities 10, 20, 30, 40 are conventional computing devices as is well known to a person skilled in the art.

The first computer entity 10 and second computer entity 20 form a trust authority hierarchy in which the first computer entity 10 acts as a root trust authority and the second computer entity 20 acts as a middle level trust authority, thereby forming a public-key infrastructure. As described in detail below, on receipt by the second computer entity 20 of a master private key generated by the first computer entity 10 the second computer entity 20 is able, using identifier-based cryptography, to generate a private/public key pair without further interaction from the first computer entity 10, where the public key can be verified, without the need for digital certificates, such that the verifier can be convinced that the public key could only be generated with knowledge of the master private key generated by the first computer entity 10.

The following embodiment utilises identifier-based cryptography using Tate pairing to provide multiple levels of trust authorities, however other types of pairing may also be used, for example Weil pairings.

For the purposes of this embodiment *G1* and *G2* denote two groups of prime order q in which the discrete logarithm problem is believed to be hard and for which there exists a computable bilinear map, for example, a Tate pairing.

i.e. $t : G_1 \times G_1 \longrightarrow G_2$

G_1 is a group of points on an elliptic curve and G_2 is a subgroup of a multiplicative group of a finite field.

As the mapping between G_1 and G_2 is bilinear exponents/multipliers can be moved around. For example if $a, b, c \in \mathbb{F}_q$ and $P, Q \in G_1$ then

$$\begin{aligned} t(aP, bQ)^c = t(aP, cQ)^b = t(bP, cQ)^a = t(bP, aQ)^c = t(cP, aQ)^b = t(cP, bQ)^a \\ = (abP, Q)^c = t(abP, cQ) = t(P, abQ)^c = t(cP, abQ) \\ = \ldots \\ = t(abcP, Q) = t(P, abcQ) = t(P, Q)^{abc} \end{aligned}$$

Additionally, for the purposes of this embodiment the following cryptographic hash functions are defined:

$$H_1 : \{0,1\}^* \longrightarrow G_1$$
$$H_2 : \{0,1\}^* \longrightarrow \mathbb{F}_q$$
$$H_3 : G_2 \longrightarrow \{0,1\}^*$$

To provide a trust hierarchy a public/private key pair is defined for a trust authority where the public key R is: $R \in$ G1 and the private key s is: $s \in \mathbb{F}_q$ with $R=sP$ where P, a public parameter, is: $P \in G_1$.

Additionally, an identifier based public key Q_{ID} / private key S_{ID} pair is defined where the Q_{ID}, $S_{ID} \in G_1$ where the trust authority's public/private key pair (R_{TA}, s) is linked with the identifier based public/private key by

$$S_{ID} = sQ_{ID} \text{ and } Q_{ID} = H_1 (ID)$$

where ID is an identifier string.

Accordingly, to allow a holder of the private part s of the trust authority's public/private key pair to sign a bit string, where m denotes the message to be signed it is necessary to compute $V = sH_1(m)$. Verification requires that the following equation is satisfied:

$$t(P, V) = t(R, H_1(m))$$

This is based upon the mapping between G_1 and G_2 being bilinear exponents/multipliers, as described above. That is to say,

$$\begin{aligned} t(P, V) &= t(P, sH_1 (m)) \\ &= t(P, H_1 (m))^s \\ &= t(sP, H_1 (m)) \\ &= t(R, H_1 (m)) \end{aligned}$$

In particular identifier based encryption allows the holder of the private key S_{ID} of an identifier based key pair to decrypt a message sent to them encrypted using the associated public key Q_{ID} .

The message to be encrypted is denoted by m.

First compute $U = rP$ where r is a random element of $\mathbb{F}_q$.

Then compute $V = m \oplus H_3(t(\mathrm{R}, rQ_{ID}))$

This results in the generation of the ciphertext U and V.

Decryption of the message is performed by computing:

$$\begin{aligned} V \oplus H3\,(t(U, S_{ID})) &= V \oplus H_3(t(rP, sQ_{ID})) \\ &= V \oplus H_3(t(P, Q_{ID})^{rs}) \\ &= V \oplus H_3(t(sP, rQ_{ID})) \\ &= V \oplus H_3(t(R, rQ_{ID})) \\ &= m \end{aligned}$$

Correspondingly identifier based signatures using Tate pairing can be implemented. For example:

First compute $r = t(P, P)^k$

where k is a random element of $\mathbb{F}_q$.

Then apply the hash function H_2 to $m\|r$ (concatenation of m and r) to obtain $h = H_2(m\|r)$.

Then compute

$$U = hS_{ID} + kP.$$

Thus generating the output U and h as the signature on the message m.

Verification of the signature can be established by computing:

$$r = t(U, P) \bullet t(Q_{ID}, R)^h$$

where the signature can only be accepted if $h = H_2(m\|r)$.

Based upon the identifier-based cryptography described above the root trust authority (i.e. the first trust authority 60) can be linked to a pseudo master private key generated by the middle level trust authority (i.e. the second trust authority 70) such that the link can be verified without the need for any digital certificates, as will now be described.

Based upon the above nomenclature table 1 lists the standard and ID based public/private key pairs that are set up for the first trust authority 60 and the second trust authority 70 where P, a pubic parameter, is an arbitrary point on an elliptic curve.

Entity	Standard Private Key	Standard Public key	ID Based Private Key	ID Based Pubic key
First TA	s_1	$R_{TA1}=s_1P$		
Second TA	s_2	$R_{TA2}=s_2P$	$S_{TA2}=s_1Q_{TA2}$	$Q_{TA2}= H_1(TA2)$

Table 1

The second trust authority 70 creates a pseudo-master private key selecting a random number r where $r \in \mathbb{F}_q$; the random number r is the pseudo-master private key. Once the pseudo-master key has been selected the second trust authority 70 generates the following public keys:

$$rs_1Q_{TA2}\ ,\ rP \text{ and } rQ_{TA2}$$

It should be noted however, that even though in the above example the second trust authority 70 has created a single pseudo-master private key the second trust authority 70 could generate any number of pseudo-master private keys.

The user 80 registers with the second trust authority 70 to obtain an associated private key for the user's public key, where the user's public key could be any form of identifier, for example the user's name 'Bob', where the public key $H_1(\text{Bob}) = Q_{\text{Bob}}$ would map to a point on an elliptic curve defined by G_1.

On registration, the second trust authority 70 provides the user 80 with the appropriate private key, which would be a combination of the user's public key and the second trust authority's pseudo private key i.e. rQ_{Bob}.

Consequently, utilizing the Tate pairing algorithms described above it is possible to verify the 'meaning' of rsQ_{TA2}, rP and rQ_{TA2} using:

$t(rP, Q_{TA2}) = t(P, rQ_{TA2})$ and
$t(P, rsQ_{TA2}) = t(sP, rQ_{TA2})$

Further (P,sP), in the above ID-based encryption and ID-based signature algorithms, can be replaced with either (P, rP) or (Q_{TA2}, rQ_{TA2}), as well as replace $t(Q_{ID}, sP) = t(sQ_{ID},P)$ with $t(Q_{Bob},rP) = t(rQ_{Bob}, P)$ or $t(Q_{Bob}, rQ_{TA2}) = t(rQ_{Bob}, Q_{TA2})$.

Figure 2 illustrates the same computer network as that shown in figure 1 with the addition of a fifth computer entity 100. The fifth computer entity 100 acts as another middle level trust authority (i.e. a third trust authority 200) independent of the second computer entity 20 where the first computer entity 10 is the root trust authority for both the second computer entity 20 and the fifth computer entity 100. As with the second computer entity 20 on receipt by the fifth computer entity 100 of a master private key generated by the first computer entity 10 the fifth computer entity 100 is able to generate a private/public key pair as described above. The network 50 could include additional middle level trust authorities, however, for the purposes of this embodiment only two middle level trust authorities will be described.

As described below, the user 80 has an independent identity associated with each middle level trust authority 70, 200, where each independent identity corresponds to a public key of the user 80. Each middle level trust authority 70, 200 provides a private key corresponding to the respective user's public

key, as described above. To send an encrypted message to the user 80 the business partner 90 encrypts the message with a combination of the user's public keys associated with the respective middle level trust authorities 70, 200 (i.e. the user's identities associated with the respective trust authorities) and the respective trust authority's public key. To recover the encrypted message the user 80 decrypts the message with a combination of the same trust authority's public keys and the user's corresponding private key.

To sign a message a user 80 uses each trust authority's public key in combination with the user's associated private keys. To verify the signature a verifier uses a combination of the trust authority's public key with the user's corresponding public keys.

The following embodiment utilises identifier-based cryptography using Tate pairings to allow the generation of a public key that is a combination of independent identities associated with respective middle level trust authorities 70, 200.

The second trust authority 70 has a public key R_{TA1} and a corresponding private key s_1 where $R_{TA1} = s_1P$, with P being a point on an elliptic curve, as described above.

The third trust authority 200 has a public key R_{TA2} and a corresponding private key s_2 where $R_{TA2} = s_2P$, with P being a point on an elliptic curve, as described above.

For n trust authorities the public/private key pair could be generalised by:

$$R_{TAi} = s_iP$$

Associated with each middle level trust authority 70, 200 the user 80 has a independent identity, that is to say with the second trust authority 70 the user 80 has an identity ID1, for example the user's name 'Bob', with third trust authority 200 the user 80 had another identity ID2, for example the name of the company the user 80 works for.

Accordingly, the user 80 has independent identity based private keys and public keys with each middle level trust authority 70, 200, where the user's identity based public key with the second trust authority 70 is $Q_{ID1} = H_1(ID1)$ and the user's identity based private key with the second trust authority 70 is S_1, where $S_1 = s_1 Q_{ID1}$ and the user's identity based public key with the third trust authority 200 is $Q_{ID2} = H_1(ID2)$ and the user's identity based private key with the third trust authority 200 is S_2, where $S_2 = s_2 Q_{ID2}$.

To allow the business partner 90 to encrypt a message m for the user 80 based upon the independent identities associated with each middle level trust authority 70, 200 the business partner 90 generates ciphertext V and U, where:

$$V = m \oplus H_3\left(\prod_{i=1}^{2} t(R_{TAi}, rQ_{IDi})\right)$$

and

$$U = rP$$

where r is a random number selected by the business partner 90.

Decryption is performed by computing:

$$m = V \oplus H_3\left(t(U, \sum_{i=1}^{2} S_i)\right)$$

Accordingly, message m can only be decrypted with knowledge of both private keys S_1, S_2.

The following embodiments utilises identifier-based cryptography using Weil pairings to allow the generation of a public key that is a combination of independent identities associated with respective middle level trust authorities 70, 200. In a more general case, the trusted authorities can be totally independent to each other and there is no needs for any business relationship to exist between the trust authorities, in fact the trust authorities do not need to know each other. For example the trust authorities may not belong to the same root trusted authority. Indeed, one or more of the trust authorities could be a root authority.

The first embodiment utilizing Weil pairings allows the business partner 90 to encrypt a message $m \in \{0,1\}^n$ for the user 80, which the user can decrypt if the user 80 has a number of private keys d_{IDi} ($i = 1, \ldots, n$), each respectively issued by a trust authority TA_i ($i = 1, \ldots, n$) corresponding to a public key Q_{IDi} ($i = 1, \ldots, n$).

Each trust authority chooses a large (at least 512-bits) prime p such that $p = 2 \bmod 3$ and $p = 6q - 1$ for some prime $q > 3$. Further, E, an elliptic curve, is defined by $y^2 = x^3 + 1$ over $\mathbb{F}_p$.

An arbitrary point on the elliptic curve is chosen, where $P \in E/\mathbb{F}_p$ of order q.

Four hash functions are defined:

$H_1: \{0,1\}^* \rightarrow \mathbb{F}_p$;

$H_2: \mathbb{F}_{p^2} \rightarrow \{0,1\}^n$ for some n;

$H_3: \{0,1\}^n \times \{0,1\}^n \rightarrow \mathbb{Z}^*_q$,

and $H_4: \{0,1\}^n \rightarrow \{0,1\}^n$.

Each trust authority TA_i (i = 1, ..., n) respectively selects a random $s_i \in \mathbb{Z}^*_q$ and set $P_{pubi} = [s_i]P$.

The user 80 registers with each respective trust authority, providing each trust authority with an appropriate independent identifier, ID_i (i = 1, ..., n) $\in \{0,1\}^*$.

Each trust authority then computes an appropriate MapToPoint ($H_1(ID_i)$) = Q_{IDi} $\in E/\mathbb{F}_p$ of order q and set the user's corresponding private key d_{IDi} to be $d_{IDi} = [s_i]Q_{IDi}$.

To encrypt a message, m, the business partner 90:

Computes a MapToPoint ($H_1(ID_i)$) = Q_{IDi} (i = 1, ..., n) $\in E/\mathbb{F}_p$ of order q.
Selects a random number $\sigma \in \{0,1\}^n$.
Computes $r = H_3(\sigma, m)$, where r is a random element that ensures only someone with the appropriate private key can decrypt the message, m.
Computes $U = [r]P$.
Computes $g_{ID} = \Pi_{(1 \le i \le n)} \hat{e}(Q_{IDi}, P_{pubi}) \in \mathbb{F}_{p^2}$.
Computes $V = \sigma \oplus H_2(g_{ID}^r)$.
Computes $W = m \oplus H_4(\sigma)$.
Sets the ciphertext to be $C = (U, V, W)$.

To decrypt the message, m, the user 80:

Tests $U \in E/\mathbb{F}_p$ of order q;
Computes $x = \hat{e}(\Sigma_{(1 \le i \le n)} d_{IDi}, U)$;
Computes $\sigma = V \oplus H_2(x)$;
Computes $m = W \oplus H_4(\sigma)$;
Computes $r = H_3(\sigma, m)$;
Checks $U = [r]P$.

The second embodiment utilizing Weil pairings allows a user 80 to sign a message, m.

The user signs a message $m \in \{0,1\}^n$ under a number of private keys d_{IDi} (i = 1, ..., n), each respectively issued by a respective trust authority, i.e. TA_i (i = 1, ..., n) corresponding to a public key Q_{IDi} (i = 1, ..., n). The business partner 90 verifies the signature by using both the user's public keys corresponding to the signing private keys and the TA_i's public keys.

As above, each trust authority choose a large (at least 512-bits) prime p such that $p = 2 \bmod 3$ and $p = 6q - 1$ for some prime $q > 3$ with E being defined by $y^2 = x^3 + 1$ over $\mathbb{F}_p$.

An arbitary point on the elliptic curve is chosen where $P \in E/\mathbb{F}_p$ of order q.

Two hash functions are chosen:
H_1: $\{0,1\}^* \rightarrow \mathbb{F}_p$;
and H_2: $\{0,1\}^n \times \{0,1\}^n \rightarrow \mathbb{Z}^*_q$.

Each trust authority TA_i (i = 1, ..., n) respectively selects a random $s_i \in \mathbb{Z}^*_q$ and set $P_{pubi} = [s_i]P$.

The user 80 registers with each respective trust authority providing each trust authority with an appropriate independent identity i.e. ID_i (i = 1, ..., n) $\in \{0,1\}^*$.

Each trust authority then computes an appropriate MapToPoint ($H_1(ID_i)$) = Q_{IDi} $\in E/\mathbb{F}_p$ of order q and sets the user's private key d_{IDi} to be $d_{IDi} = [s_i]Q_{IDi}$.

To sign a message, m, the user 80:

Selects a random $z \in \{0,1\}^n$;

Computes $U = [z]P$;

Computes $h = H_2(m, U)$;

Computes $V = [h] \Sigma_{(1 \le i \le n)} d_{IDi} + [z] \Sigma_{(1 \le i \le n)} P_{pubi}$

Ships to the business partner m, U and V.

To verify the signature (m, U, V) the business partner 90:

Computes MapToPoint ($H_1(ID_i)$) = $Q_{IDi} \in E/\mathbb{F}_p$ of order q;

Computes $h = H_2(m, U)$;

Computes $x = \hat{e}(P, V)$;

Computes $y = \Pi_{(1 \le i \le n)} \hat{e}(P_{pubi}, [h]Q_{IDi} + U)$;

Checks $x == y$.

The third embodiment utilizing Weil pairing provides a further embodiment that allows a user 80 to sign a message.

The user 80 signs a message $m \in \{0,1\}^n$ under a number of private keys d_{IDi} (i = 1, ..., n), each respectively issued by a respective trust authority i.e. TA_i (i = 1, ..., n) corresponding to a public key Q_{IDi} (i = 1, ..., n). The business partner 90 verifies the signature by using both the user's public keys corresponding to the signing private keys and the TA_i's public keys.

As above, each trust authority choose a large (at least 512-bits) prime p such that $p = 2 \bmod 3$ and $p = 6q - 1$ for some prime $q > 3$ with E being defined by $y^2 = x^3 + 1$ over $\mathbb{F}_p$.

An arbitrary point P on the elliptic curve is chosen, where $P \in E/\mathbb{F}_p$ of order q.

Two hash functions are chosen:

H_1: $\{0,1\}^* \rightarrow \mathbb{F}_p$;

and H_2: $\{0,1\}^n \times \{0,1\}^n \rightarrow \mathbb{Z}^*_q$.

Each trust authority TA_i (i = 1, ..., n) respectively selects a random $s_i \in \mathbb{Z}^*_q$ and set $P_{pubi} = [s_i]P$.

The user 80 registers with each respective trust authority providing each trust authority with an appropriate independent identity i.e. ID_i (i = 1, ..., n) $\in \{0,1\}^*$.

Each trust authority computes an appropriate MapToPoint ($H_1(ID_i)$) = $Q_{IDi} \in E/\mathbb{F}_p$ of order q and sets the private key d_{IDi} to be $d_{IDi} = [s_i]Q_{IDi}$.

To sign a message, m, the user 80:

Selects a random $k \in \{0,1\}^n$;
Computes $e = \hat{e}(\Sigma_{(1 \le i \le n)} d_{IDi}, P)$;
Computes $r = e^k$;
Computes $h = H_2(m, r)$;
Computes $S = ([k] - [h]) \Sigma_{(1 \le i \le n)} d_{ID}$;
Ships to the business partner m, h and S.

Verify the signature (m, h, S) the business partner 90:

Computes MapToPoint ($H_1(ID_i)$) = $Q_{IDi} \in E/\mathbb{F}_p$ of order q;
Computes $e' = \Pi_{(1 \le i \le n)} \hat{e}(Q_{IDi}, P_{pubi})$ – may be precomputed;
Computes $r' = \hat{e}(S, P)e'^h$;
Checks $h == H_2(m, r')$.

CLAIMS

1. Method for generating a private key comprising generating a first and second cryptographic key for a first party; generating a third and fourth cryptographic key for a second party wherein the fourth cryptographic key is derived from the first and third cryptographic key; generating a number that in association with the second cryptographic key, the third cryptographic key and the fourth cryptographic key define a first cryptographic parameter, a second cryptographic parameter and a third cryptographic parameter respectively; combining the number with a third party's public key to define an associated private key.

2. Method for generating a cryptographic key comprising generating a first cryptographic key and a second cryptographic key for a first party; generating a third cryptographic key and fourth cryptographic key for a second party wherein the fourth cryptographic key is derived from the first cryptographic key and third cryptographic key; generating a number that in association with the second cryptographic key, the third cryptographic key and the fourth cryptographic key define a first, second and third cryptographic parameter respectively; combining the number with a fifth cryptographic key associated with a third party to define an associated cryptographic key such that an association can be established between the fifth cryptographic key of the third party and the second cryptographic key of the first party.

3. Method for generating a private key comprising generating a first private key and public key for a first party; generating a second private and public key for a second party wherein the second private key is derived from the first private key and second public

key; generate a number that in association with the first pubic key, the second private and public key define a first, second and third public parameter respectively; combining the number with a third pubic key associated with a third party to define an associated private key such that an association can be established between the third public key of the third party and the first public key of the first party.

4. Method according to claim 3, wherein the number is a random number.

5. Method according to claim 3 or 4, wherein the association between the third public key and first public key is established using a bilinear map.

6. Method according to claim 5, wherein the bilinear map is either a Tate or Weil pairing.

7. Method according to any of claims 3 to 6, wherein the first party is a first trusted party and the second party is a second trusted party.

8. Method for generating a private key comprising generating a first private key and public key for a first party; generating a second private and public key for a second party wherein the second private key is derived from the first private key and second public key; generate a third private key for the second party that in association with the first public key, the second private key and the second public key define a first cryptographic parameter, a second cryptographic parameter and a third public key respectively; combining the third private key with a third party's public key to define an associated private key such that an association can be

established between the third public key of the second party and the first public key of the first party.

9. Computer apparatus for generating a private key comprising a processor arranged to generate a number that in association with a first private key and public key associated with a first party define a first and second public parameter respectively wherein the first private key is derived from a second private key associated with a second party and the first public key; and combining the number with a second public key associated with a third party to define an associated private key such that an association can be established between the second public key of the third party and a third public key of the second party.

10. Computer apparatus according to claim 8, wherein the number is a random number.

11. Computer apparatus according to claim 8 or 9, wherein the association between the second public key and the third public key is established using a Tate or Weil pairing.

12. Computer apparatus according to any of claims, wherein the first party is a first trusted party and the second party is a second trusted party.

ABSTRACT

METHOD AND APPARATUS FOR GENERATING A CRYPTOGRAPHIC KEY

Method for generating a cryptographic key comprising generating a first cryptographic key and a second cryptographic key for a first party; generating a third cryptographic key and fourth cryptographic key for a second party wherein the fourth cryptographic key is derived from the first cryptographic key and third cryptographic key; generating a number that in association with the second cryptographic key, the third cryptographic key and the fourth cryptographic key define a first, second and third cryptographic parameter respectively; combining the number with a fifth cryptographic key associated with a third party to define an associated cryptographic key such that an association can be established between the fifth cryptographic key of the third party and the second cryptographic key of the first party.

Figure 1

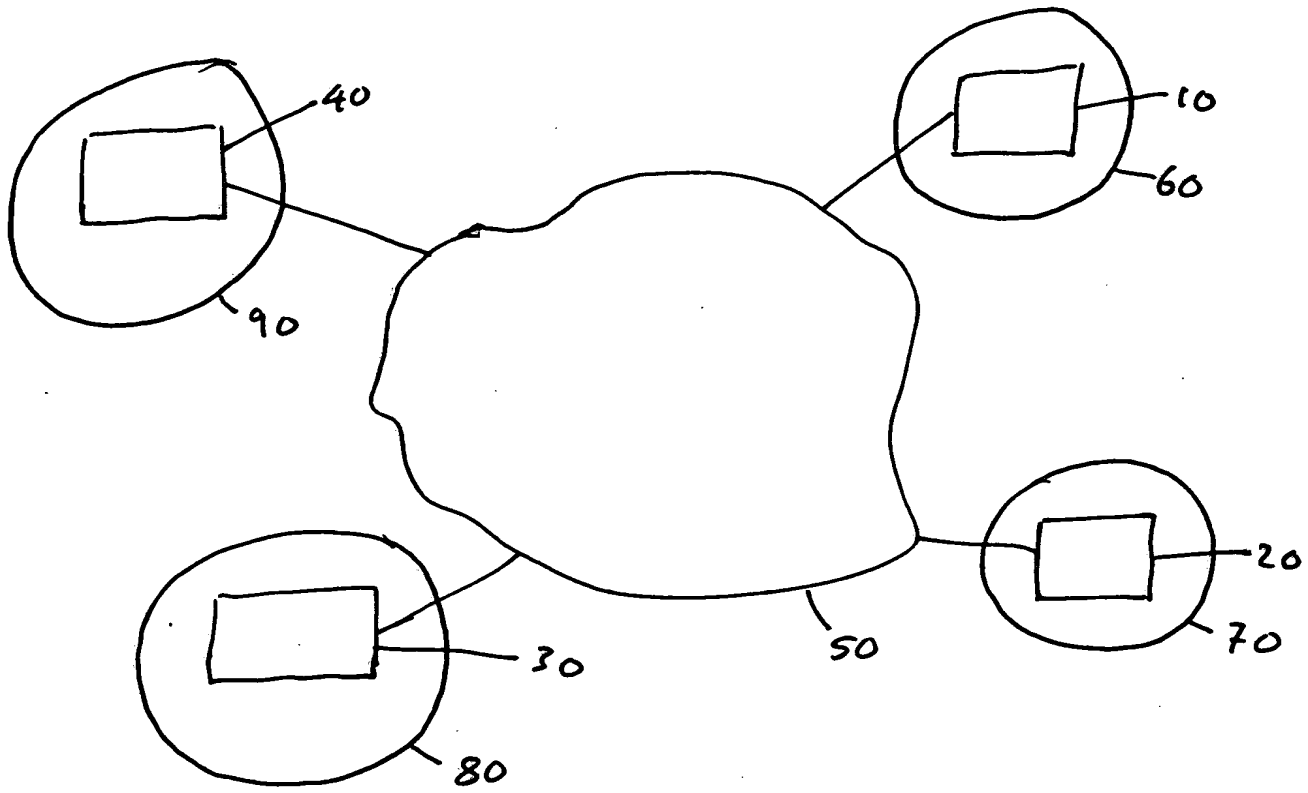


Figure 1

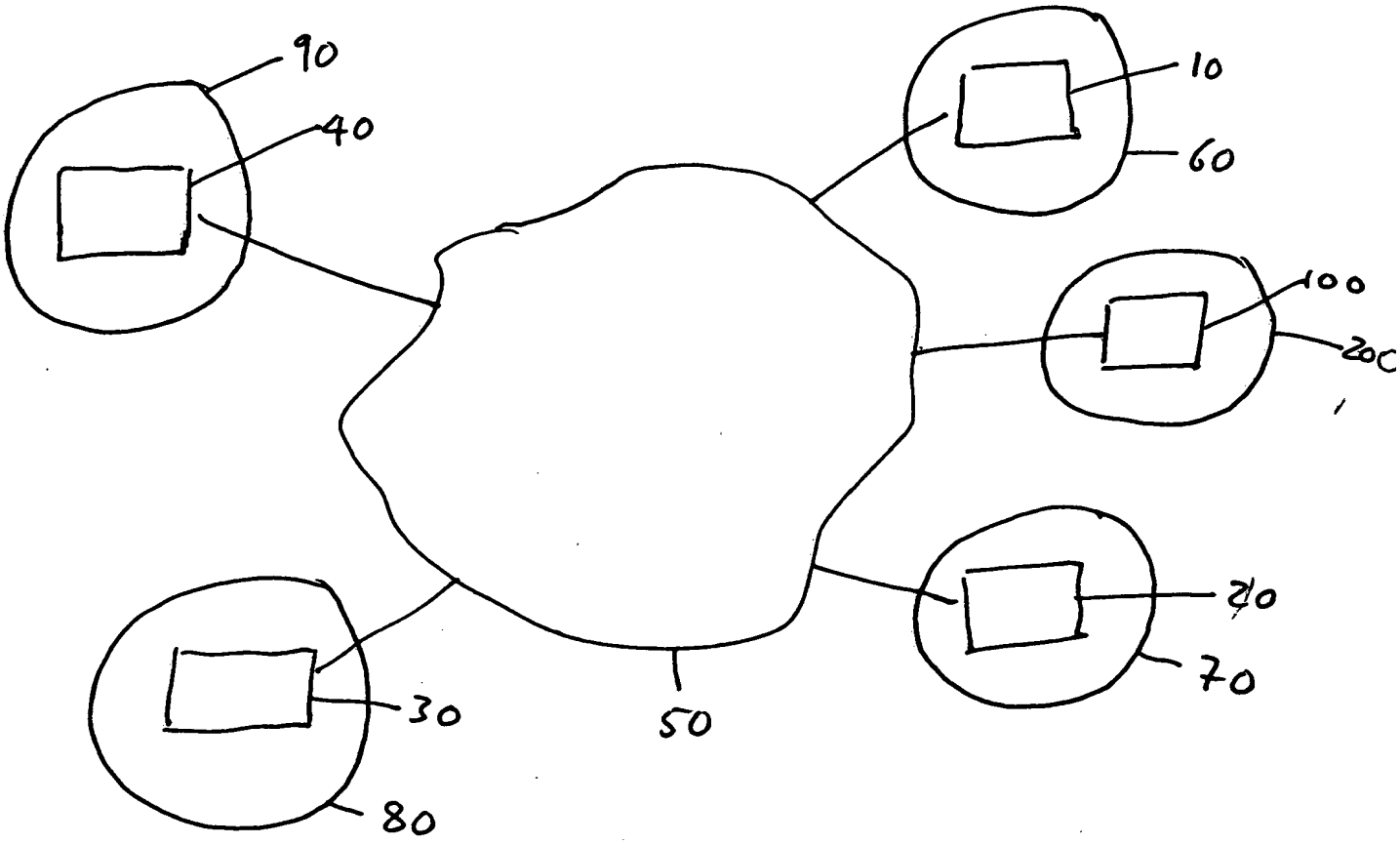


Figure 2